International Journal on Applications in Information and Communication Engineering
Volume 5 : Issue 1: May 2019, pp 28 – 34 www.aetsjournal.com ISSN (Online) : 2394 - 6237

AN EFFICIENT TO IMPROVE SECURITY TASK ALLOCATION USING MOBILE CLOUD COMPUTING

S.MANI PRIYA , Dr.K.MUNUSAMY

Abstract— In cloud storage services, users store their data remotely to the cloud and realize the data sharing with others. Remote data integrity auditing is introduced to guarantee the data integrity in the cloud storage. In Electronic Health Records (EHRs) system, the cloud file might contain some sensitive information. The sensitive information should not be known to others when the cloud file is shared. Encrypting the whole shared file realizes the sensitive information hiding, but will make this shared file unable to be used by others. How to realize data sharing with sensitive information hiding in remote data integrity auditing still has not been explored up to now. In order to address this problem, this project proposes a remote data integrity auditing scheme that realizes data sharing with sensitive information hiding. Moreover, users should be able to just use the cloud storage as if it is local, without worrying about the need to verify its integrity. Thus, enabling public auditability for cloud storage is of critical importance so that users can resort to a third party auditor (TPA) to check the integrity of outsourced data and be worry-free. To securely introduce an effective TPA, the auditing process should bring in no new vulnerabilities towards user data privacy, and introduce no additional online burden to user. This project proposes a secure cloud storage system supporting privacy-preserving public auditing. We further extend our result to enable the TPA to perform audits for multiple users simultaneously and efficiently. In addition, it articulates performance optimization mechanisms for this scheme, and in particular present an efficient method for selecting optimal parameter values to minimize the computation costs of clients and storage service providers. It shows that the solution introduces lower computation and communication overheads in comparison with non-cooperative approaches.

I. INTRODUCTION

Cloud computing creates a large number of security issues and challenges. A list of security threats to cloud computing is presented . These issues range from the required trust in the cloud provider and attacks on cloud interfaces to misusing the cloud services for attacks on other systems.

Mani Priya.S , Department of Computer Science Engineering , J.K.K Nattraja College of Engineering and Technology , Komarapalayam. (Email ID : manipriyasakthi06@gmail.com)
Dr.K.Munusamy M.E., Ph.D., Assistant Professor , Department of Computer Science Engineering , J.K.K Nattraja College of Engineering and Technology , Komarapalayam.
(Email ID : K.munusamy65@gmail.com)

The main problem that the cloud computing paradigm implicitly contains is that of secure outsourcing of sensitive as well as business-critical data and processes. When considering using a cloud service, the user must be aware of the fact that all data given to the cloud provider leave the own control and protection sphere. Even more, if deploying data-processing applications to the cloud (via IaaS or PaaS), a cloud provider gains full control on these processes. Hence, a strong trust relationship between the cloud provider and the cloud user is considered a general prerequisite in cloud computing. Depending on the political context this trust may touch legal obligations. For instance, Italian legislation requires that government data of Italian citizens, if collected by official agencies, have to remain within Italy.

Thus, using a cloud provider from outside of Italy for realizing an e-government service provided to Italian citizens would immediately violate this obligation. Hence, the cloud users must trust the cloud provider hosting their data within the borders of the country and never copying them to an off-country location (not even for backup or in case of local failure) nor providing access to the data to entities from abroad.

An attacker that has access to the cloud storage component is able to take snapshots or alter data in the storage. This might be done once, multiple times, or continuously. An attacker that also has access to the processing logic of the cloud can also modify the functions and their input and output data.

Replication of applications allows to receive multiple results from one operation performed in distinct clouds and to compare them within the own premise. This enables the user to get an evidence on the integrity of the result.

Partition of application System into tiers allows separating the logic from the data. This gives additional protection against data leakage due to flaws in the application logic.

Partition of application logic into fragments allows distributing the application logic to distinct clouds. This has two benefits. First, no cloud provider learns the complete application logic. Second, no cloud provider

learns the overall calculated result of the application. Thus, this leads to data and application confidentiality.

Partition of application data into fragments allows distributing fine-grained fragments of the data to distinct clouds. None of the involved cloud providers gains access to all the data, which safeguards the data's confidentiality.

Each of the introduced architectural patterns provides individual security merits, which map to different application scenarios and their security needs. Obviously, the patterns can be combined resulting in combined security merits, but also in higher deployment and runtime effort. The following sections present the four patterns in more detail and investigate their merits and flaws with respect to the stated security requirements under the cloud.

II. LITERATURE SURVEY

1)EXPLORING INFORMATION LEAKAGE IN THIRD-PARTY COMPUTE CLOUDS

The authors stated that third-party cloud computing represents the promise of outsourcing as applied to computation. Services, such as Microsoft's Azure and Amazon's EC2, allow users to instantiate virtual machines (VMs) on demand and thus purchase precisely the capacity they require when they require it. In turn, the use of virtualization allows third-party cloud providers to maximize the utilization of their sunk capital costs by multiplexing many customer VMs across a shared physical infrastructure. However, in this paper, the authors showed that this approach can also introduce new vulnerabilities. Using the Amazon EC2 service as a case study, they showed that it is possible to map the internal cloud infrastructure, identify where a particular target VM is likely to reside, and then instantiate new VMs until one is placed co-resident with the target. They explored how such placement can then be used to mount cross-VM side-channel attacks to extract information from a target VM on the same machine.It has become increasingly popular to talk of "cloud computing" as the next infrastructure for hosting data and deploying software and services. In addition to the plethora of technical approaches associated with the term, cloud computing is also used to refer to a new business model in which core computing and software capabilities are outsourced on demand to shared third-party infrastructure.

While this model, exemplified by Amazon's Elastic Compute Cloud (EC2), Microsoft's Azure Service Platform, and Rackspace's Mossoprovides a number of advantages including economies of scale, dynamic provisioning, and low capital expendituresit also introduces a range of new risks. Some of these risks are self-evident and relate to the new trust relationship between customer and cloud provider. For example, customers must trust their cloud providers to respect the privacy of their data and the integrity of their computations. However, cloud infrastructures can also introduce non-obvious threats from other customers due to the subtleties of how physical resources can be transparently shared between virtual machines (VMs).In particular, to maximize efficiency multiple VMs may be simultaneously assigned to execute on the same physical server. Moreover, many cloud providers allow "multi-tenancy" multiplexing the virtual machines of disjoint customers upon the same physical hardware. Thus it is conceivable that a customer's VM could be assigned to the same physical server as their adversary. This in turn, engenders a new threat that the adversary might penetrate the isolation between VMs (e.g., via a vulnerability that allows an "escape" to the hypervisor or via side-channels between VMs) and violate customer confidentiality. This paper explores the practicality of mounting such cross-VM attacks in existing third-party compute clouds. The attacks they considered require two main steps: placement and extraction. Placement refers to the adversary arranging to place their malicious VM on the same physical machine as that of a target customer.

Using Amazon's EC2 as a case study, they demonstrated that careful empirical "mapping"can reveal how to launch VMs in a way that maximizes the likelihood of an advantageous placement. They found that in some natural attack scenarios, just a few dollars invested in launching VMs can produce a 40% chance of placing a malicious VM on the same physical server as a target customer. Using the same platform they also demonstrated the existence of simple, low-overhead, "co-residence" checks to determine when such an advantageous placement has taken place. While they focused on EC2, they believed that variants of our techniques are likely to generalize to other services, such as Microsoft's Azure or Rackspace's Mosso, as they only utilized standard customer capabilities and do not require that cloud providers disclose details of their infrastructure or assignment policies.Having managed to place a VM co-resident with the target, the next step is to extract confidential information via a cross-VM attack. While there are a number of avenues for such an attack, in this paper we focus on side-channels: cross-VM information leakage due to the sharing of physical resources (e.g., the CPU's data caches). In the multi-

process environment, such attacks have been shown to enable extraction of RSA and AES secret keys. However, they are unaware of published extensions of these attacks to the virtual machine environment; indeed, there are significant practical challenges in doing soThey showed preliminary results on cross-VM side channel attacks, including a range of building blocks (e.g., cache load measurements in EC2) and coarse-grained attacks such as measuring activity burst timing (e.g., for cross-VM keystroke monitoring). This point to the practicality of side-channel attacks in cloud-computing environments. Overall, their results indicated that there exist tangible dangers when deploying sensitive tasks to third-party compute clouds.

2) CROSS-VM SIDE CHANNELS AND THEIR USE TO EXTRACT PRIVATE KEYS YINQIAN ZHANG and ARI JUELS

The author details the construction of an access-driven side-channel attack by which a malicious virtual machine (VM) extracts fine-grained information from a victim VM running on the same physical computer. This attack is the first such attack demonstrated on a symmetric multiprocessing system virtualized using a modern VMM (Xen). Such systems are very common today, ranging from desktops that use virtualization to sandbox application or OS compromises, to clouds that co-locate the workloads of mutually distrustful customers. Constructing such a side-channel requires overcoming challenges including core migration, numerous sources of channel noise, and the difficulty of preempting the victim with sufficient frequency to extract fine-grained information from it. This paper addresses these challenges and demonstrates the attack in a lab setting by extracting an ElGamal decryption key from a victim using the most recent version of the libgcrypt cryptographic library.

Modern virtualization technologies such as Xen, HyperV, and VMWare are rapidly becoming the cornerstone for the security of critical computing systems. This reliance stems from their seemingly strong isolation guarantees, meaning their ability to prevent guest virtual machines (VMs) running on the same system from interfering with each other's execution or, worse, exfiltrating confidential data across VM boundaries. The assumption of strong isolation underlies the security of public cloud computing systems such as Amazon EC2, Microsoft Windows Azure, and Rackspace; military multi-level security environments; home user and enterprise desktop security in the face of compromise ; and software-based trusted computing. VM managers (VMMs) for modern virtualization systems attempt to realize this assumption by enforcing logical isolation between VMs using traditional access-control mechanisms. But such logical isolation may not be sufficient if attackers can circumvent them via side-channel attacks. Concern regarding the existence of such attacks in the VM setting stems from two facts. First, in non-virtualized, cross-process isolation contexts, researchers have demonstrated a wide variety of side-channel attacks that can extract sensitive data such as cryptographic keys on single-core architectures. The most effective attacks tend to be so-called "access-driven" attacks that exploit shared microarchitectural components such as caches.

Second, Ristenpart et al.exhibited coarser, cross-VM, access-driven side-channel attacks on modern symmetric multi-processing (SMP, also called multi-core) architectures. But their attack could only provide crude information (such as aggregate cache usage of a guest VM) and, in particular, is insufficient for extracting cryptographic secrets. Despite the clear potential for attacks, no actual demonstrations of fine-grained cross-VMside-channels attacks have appeared. The oft-discussed challenges to doing so stem primarily from the facts that VMMs place more layers of isolation between attacker and victim than in cross-process settings, and that modern SMP architectures do not appear to admit fine-grained side-channel attacks (even in non-virtualized settings) because the attacker and victim are often assigned to disparate cores. Of course a lack of demonstrated attack is not a proof of security, and so whether fine-grained cross-VM side-channel attacks are possible has remained an important open question. In this paper, the authors presented the development and application of a cross-VM side-channel attack in exactly such an environment. Like many attacks before, theirs an access- driven attack in which the attacker VM alternates execution with the victim VM and leverages processor caches to observe behavior of the victim. However, they believed many of the techniques we employ to accomplish this effectively and with high fidelity in a virtualized SMP environment are novel. In particular, they provided an account of how to overcome three classes of significant challenges in this environment:

Finally, they customized their attack to the task of extracting a private decryption key from the victim and specifically show how to "stitch together" these intermittent, partial observations of the victim VM activity to assemble an entire private key. As they demonstrated in a lab testbed, their attack establishes a

side-channel of sufficient fidelity that an attacker VM can extract a private ElGamal decryption key from a co-resident victim VM running Gnu Privacy Guard (GnuPG), a popular software package that implements the OpenPGP e-mail encryption standard. The underlying vulnerable code actually lies in the most recent version of the libgcrypt library, which is used by other applications and deployed widely.

Specifically, they showed that the attacker VM's monitoring of a victim's repeated exponentiations over the course of a few hours provides it enough information to reconstruct the victim's 457-bit private exponent accompanying a 4096-bit modulus with very high accuracy—so high that the attacker was then left to search fewer than 10, 000 possible exponents to find the right one.

They stressed, moreover, that much about their attack generalizes beyond ElGamal decryption (or, more generally, discovering private exponents used in modular exponentiations) in libgcrypt. In particular, their techniques for preempting the victim frequently for observation and sidestepping several sources of cache noise are independent of the use to which the side-channel is put. Even those components that they necessarily tune toward ElGamal private-key extraction, and the pipeline of components overall, should provide a roadmap for constructing side-channels for other ends. They thus believed that their work serves as a cautionary note for those who rely on virtualization for guarding highly sensitive secrets of many types, as well as motivation for the research community to endeavor to improve the isolation properties that modern VMMs provide to a range of applications.

3) SECURITY ANALYSIS OF CLOUD MANAGEMENT INTERFACES SECURIJURAJ SOMOROVSKY and MARIO HEIDERICH

The authors stated that Cloud Computing resources are handled through control interfaces. It is through these interfaces that the new machine images can be added, existing ones can be modified, and instances can be started or ceased. Effectively, a successful attack on a Cloud control interface grants the attacker a complete power over the victim's account, with all the stored data included. In this paper, the authors provided a security analysis pertaining to the control interfaces of a large Public Cloud (Amazon) and widely used Private Cloud software (Eucalyptus).Their research results are alarming: in regards to the Amazon EC2 and S3 services, the control interfaces could be compromised via the novel signature wrapping and advanced XSS techniques. Similarly, the Eucalyptus control interfaces were vulnerable to classical signature wrapping attacks, and had nearly no protection against XSS. As a follow up to those discoveries, they additionally describes the ountermeasures against these attacks, as well as introduce a novel "black box" analysis methodology for public Cloud interfaces.

The cloud computing paradigm has been hailed for its promise of enormous cost-saving potential. In spite of this euphoria, the consequences regarding a migration to the cloud need to be thoroughly considered. Amongst many obstacles present, the highest weight is assigned to the issues arising within security. Cloud security discussions to date mostly focus on the fact that customers must completely trust their cloud providers with respect to the confidentiality and integrity of their data, as well as computation faultlessness. However, another important area is often overlooked: if the Cloud control interface is compromised, the attacker gains immense potency over the customer's data. This attack vector is a novelty as the result of the control interface (alongside with virtualization techniques) being a new feature of the Cloud Computing paradigm, as NIST lists On-demand self-service and Broad network access as essential characteristics of Cloud Computing systems.

In this paper, the authors refer to two distinct classes of attacks on the two main authentication mechanisms used in Amazon EC2 and Eucalyptus cloud control interfaces. The first class of attacks complies of the XML Signature Wrapping attacks (or in short { signature wrapping attacks) on the public SOAP interface of the Cloud. They demonstrated that these control interfaces are highly vulnerable to several new and classical variants of signature wrapping. For these attacks, knowledge of a single signed SOAP message is sufficient to attain a complete compromization of the security within the customer's account. The reason for this easiness is that one can generate arbitrary SOAP messages accepted by this interface from only one valid signature. To make things even worse, in one attack variant, knowledge of the (public) X.509 certificate alone enabled a successful execution of an arbitrary cloud control operation on behalf of the certificate owner.

Those included actions such as starting or stopping virtual machines, downloading or uploading virtual machine image files, resetting the administrator's password for cloud instances, and so on. The second class are advanced XSS attacks on browser based Web front-ends.

They found a persistent Cross Site Scripting (XSS) vulnerability that allowed an adversary to perform an automated attack targeted at stealing username/password data from EC2/S3 customers. This attack was made possible by the simple fact the Amazon shop and the Amazon cloud control interfaces share the same log-in credentials, thus any XSS attack on the (necessarily complex) shop interface can be turned into an XSS attack on the cloud control interface. The Eucalyptus Web front-end was equally prone to these kind of attacks. Their analysis has shown that in order to compromise this system, the attacker could easily use a simple HTML injection.

From a conceptual standpoint, cloud services need some form of cloud control which enables users to manage and configure the service, whilst also preserving access to the stored data. In IaaS-based clouds the control interface allows to, for example, instantiate machines, as well as to start, pause and stop them. Machine images can be created or modified, and the links to persistent storage devices must be configured. It is therefore quite undebatable that the security of a cloud service highly depends on robust and effective security mechanisms for the cloud control interfaces.

Technically, the cloud control interface can be realized either as a SOAP-based Web Service, or as a Web application. If the control interface is SOAP-based, then WS-Security can be applied to provide security services. For the authentication purposes, security tokens (mainly X.509 certificates) and XML Signature can be employed. A problem that generally arises is that the WS-Security standard is vulnerable to signature wrapping attacks, which consequently may invalidate this authentication mechanism.

4) AMAZONIA:WHEN ELASTICITY SNAPS BACK SVEN BUGIEL and STEFAN NÜRNBERGER

Describe cloud Computing is an emerging technology promising new business opportunities and easy deployment of web services. Much has been written about the risks and benefits of cloud computing in the last years. The literature on clouds often points out security and privacy challenges as the main obstacles, and proposes solutions and guidelines to avoid them. However, most of these works deal with either malicious cloud providers or customers, but ignore the severe threats caused by unaware users.In this paper they considered security and privacy aspects of real-life cloud deployments, independently from malicious cloud providers or customers. They focused on the popular Amazon Elastic Compute Cloud (EC2) and give a detailed and systematic analysis of various crucial vulnerabilities in publicly available and widely used Amazon Machine Images (AMIs) and show how to eliminate them. Their Amazon Image Attacks (AmazonIA) deploy an auto- mated tool that uses only publicly available interfaces and makes no assumptions on the underlying cloud infrastructure. They were able to extract highly sensitive information (including passwords, keys, and credentials) from a variety of publicly available AMIs. The extracted information allows to

- start (botnet) instances worth thousands of dollars per day,
- provide backdoors into the running machines,
- launch impersonation attacks, or
- access the source code of the entire web service.

Their attacks can be used to completely compromise several real web services offered by companies (including IT-security companies), e.g., for website statistics/user tracking, two-factor authentication, or price comparison. Further, they showed mechanisms to identify the AMI of certain running instances.

Following the maxim “security and privacy by design" they showed how their automated tools together with changes to the user interface can be used to mitigate their attacks. Cloud computing offers fine-grained IT resources, including storage, networking, and computing platforms, on an on-demand and pay-per-use basis. The high usability of today's cloud computing platforms makes this rapidly emerging paradigm very attractive for customers who want to instantly and easily provide web-services that are highly available and scalable to the current demands.

In the most flexible service model of cloud computing, Infrastructure-as-a-Service (IaaS), customers can build entire virtual infrastructures by renting resources like storage, network, and computing platforms in form of virtual machines with administrative access to the whole operating system. Images of these virtual machines can easily be shared to be run by other users, similar to an app store for the cloud.Albeit the various advantages of cloud computing, serious concerns about security and privacy hinder many users from “going into the cloud". Most solutions to preserve security and privacy in the cloud proposed so far consider potentially faulty/malicious cloud providers or technical savvy/rogue customers. However, the much more serious and ubiquitous threat of unaware users who unintentionally harm their own or others' security or privacy is often overseen.

The main goal of this paper is the investigation and evaluation of security and privacy threats caused by the unawareness of users in the cloud. Although the methods and techniques described in this paper are applicable to arbitrary IaaS providers, they focused on one of the major cloud providers, Amazon's Elastic Compute Cloud (EC2) and adapt their terminology accordingly. In the following, they described the players involved in the (Amazon) Cloud App Store and the resulting security challenges.

III. PROPOSED SYSTEM

The proposed system includes all the existing system approach which covers multiple cloud service provider environments. In addition, size blocks of data are being processed with varying size nature in different cloud locations having same copy of data. The data blocks is stored and retrieved in different cloud locations based on the storage and computational capability. Thus the proposed system explores such issue to provide the support of variable-length block verification. Likewise, the privacy level for all cloud providers is analyzed by trusted authority and security degree and performance is quantified for encryption algor ithms.

ADVANTAGES

The proposed system has following advantages.

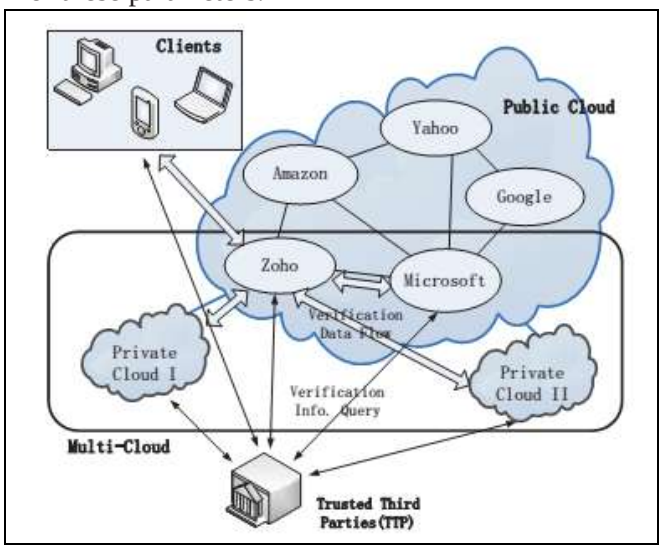
- Partial data of files are taken from multiple mirror locations and send to selected client.
- Suitable for very large size files.
- Irrelevant size blocks of data are handled among the multiple cloud service providers based on their computational capabilities.
- Different trust level is set to different cloud providersand encryption/decryption is varied based on the clouds computational capability.

IV. BLOCK DIAGROM

CP-PDP (Cooperative Provable Data Possession for Integrity Verification) schemes are incapable to satisfy the inherent requirements from multiple clouds in terms of communication and computation costs.

In this architecture, a data storage service involves three different entities:

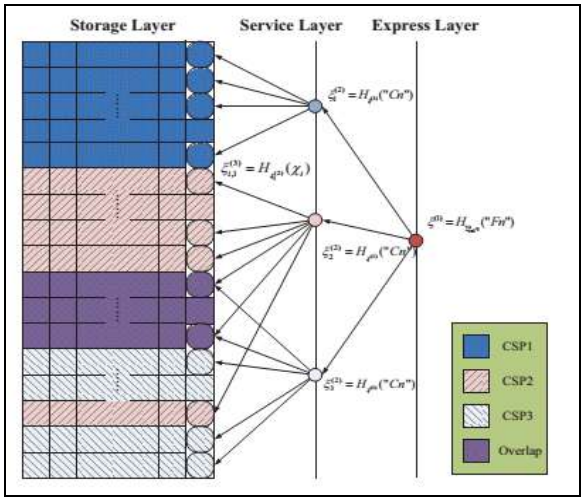
- Clients who have a large amount of data to be stored in multiple clouds and have the permissions to access and manipulate stored data;
- Cloud Service Providers (CSPs) who work together to provide data storage services and have enough storages and computation resources;
- Trusted Third Party (TTP) who is trusted to store verification parameters and offer public query services for these parameters.



In this proposed methodology make use of this simple hierarchy to organize data blocks from multiple CSP services into a large size file by shading their differences among these cloud storage systems

This hierarchical structure $\mathcal{H}$ consists of three layers to represent relationships among all blocks for stored resources. They are described as follows:

- Express Layer: offers an abstract representation of the stored resources.
- Service Layer: offers and manages cloud storage services.
- Storage Layer: realizes data storage on many physical devices.



References

[1] T. Ristenpart, E. Tromer, H. Shacham, and S. Savage, "Hey, You, Get Off of My Cloud: Exploring Information Leakage in Third-Party Compute Clouds," Proc. 16th ACM Conf. Computer and Comm. Security (CCS '09), pp. 199-212, 2016.

[2] Y. Zhang, A. Juels, M.K.M. Reiter, and T. Ristenpart, "Cross-VM Side Channels and Their Use to Extract Private Keys," Proc. ACM Conf. Computer and Comm. Security (CCS '12), pp. 305-316, 2015.

[3] J. Somorovsky, M. Heiderich, M. Jensen, J. Schwenk, N. Gruschka, and L. Lo Iacono, "All Your Clouds Are Belong to Us: Security Analysis of Cloud Management Interfaces," Proc. Third ACM Workshop Cloud Computing Security Workshop (CCSW '11), pp. 3-14, 2014.

[4] S. Bugiel, S. Nu ¨ rnberger, T. Po ¨ppelmann, A.-R. Sadeghi, and T.Schneider, "AmazonIA: When Elasticity Snaps Back," Proc. 18th ACM Conf. Computer and Comm. Security (CCS '11), pp. 389-400, 2011.

[5] G. Danezis and B. Livshits, "Towards Ensuring Client-Side Computational Integrity (Position Paper)," Proc. ACM Cloud Computing Security Workshop (CCSW '11), pp. 125-130, 2011.

[6] S. Grob and A. Schill, "Towards User Centric Data Governance and Control in the Cloud," Proc. IFIP WG 11.4 Int'l Conf. Open Problems in Network Security (iNetSeC), pp. 132-144, 2011.

[7] M. Burkhart,M. Strasser, D. Many, and X. Dimitropoulos, "SEPIA: Privacy-Preserving Aggregation of Multi-Domain Network Events and Statistics," Proc. USENIX Security Symp., pp. 223-240, 2010.

[8] D. Hubbard and M. Sutton, "Top Threats to Cloud Computing V1.0," Cloud Security Alliance, http://www. cloudsecurityalliance.org/ top threats, 2010.

[9] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia. A view of cloud computing. Commun. ACM, 53(4):50–58, 2010.

[10] R. Meushaw and D. Simard. A network on a desktop. NSA Tech Trend Notes, 9(4), 2000. http://www.vmware.com/pdf/TechTrendNotes.pdf.